Universidade do Porto
Faculdade de Engenharia
FEUP

Crawling, Curating and Pre-processing Tweets

sapo.pt

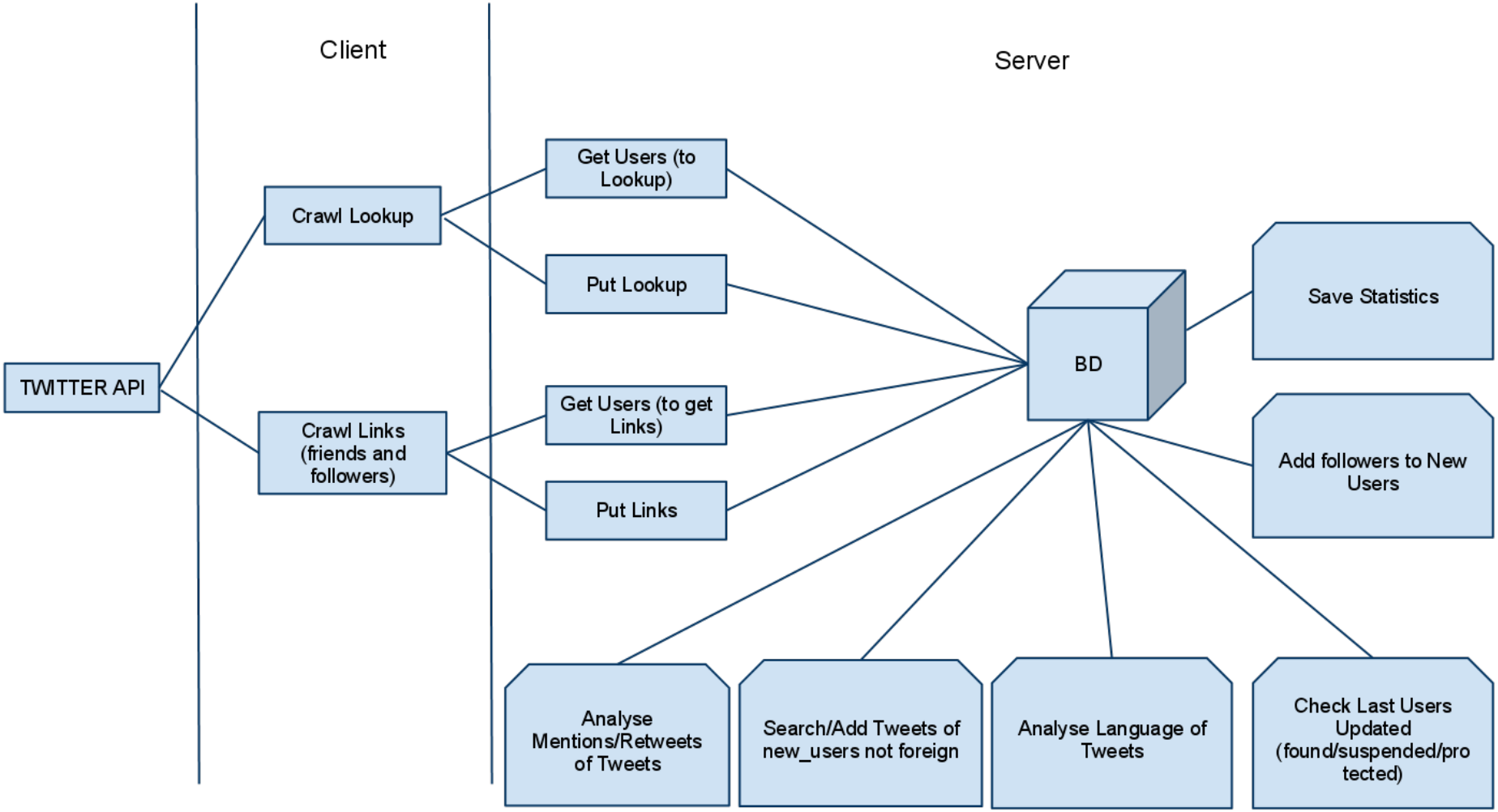
Team

- Eduardo Oliveira (Msc Std):
  - Crawling Server Side
- José Martins (Msc Std):
  - Crawling Client Side
- Matko Bosniak (Reaction Scholarship):
  - Data Curation and Network Analysis
- Gustavo Laboreiro (PhD Student):
  - Pre-Processing of UGC
- Eduarda Rodrigues + Luís Sarmento + Eugénio Oliveira
  - Supervision

Crawling

Eduardo Oliveira
José Martins

Overview


Client
Server
TWITTER API
Crawl Lookup
Crawl Links (friends and followers)
Get Users (to Lookup)
Put Lookup
Get Users (to get Links)
Put Links
BD
Save Statistics
Add followers to New Users
Analyse Mentions/Retweets of Tweets
Search/Add Tweets of new_users not foreign
Analyse Language of Tweets
Check Last Users Updated (found/suspended/protected)

Client

- Simple slave application focused on getting data
  - Processing is left to the server
- Has an application key requested to Twitter
- Authenticates using Oauth
- Performs accesses to the Twitter API, while trying to be well under the traffic limits

Two Clients

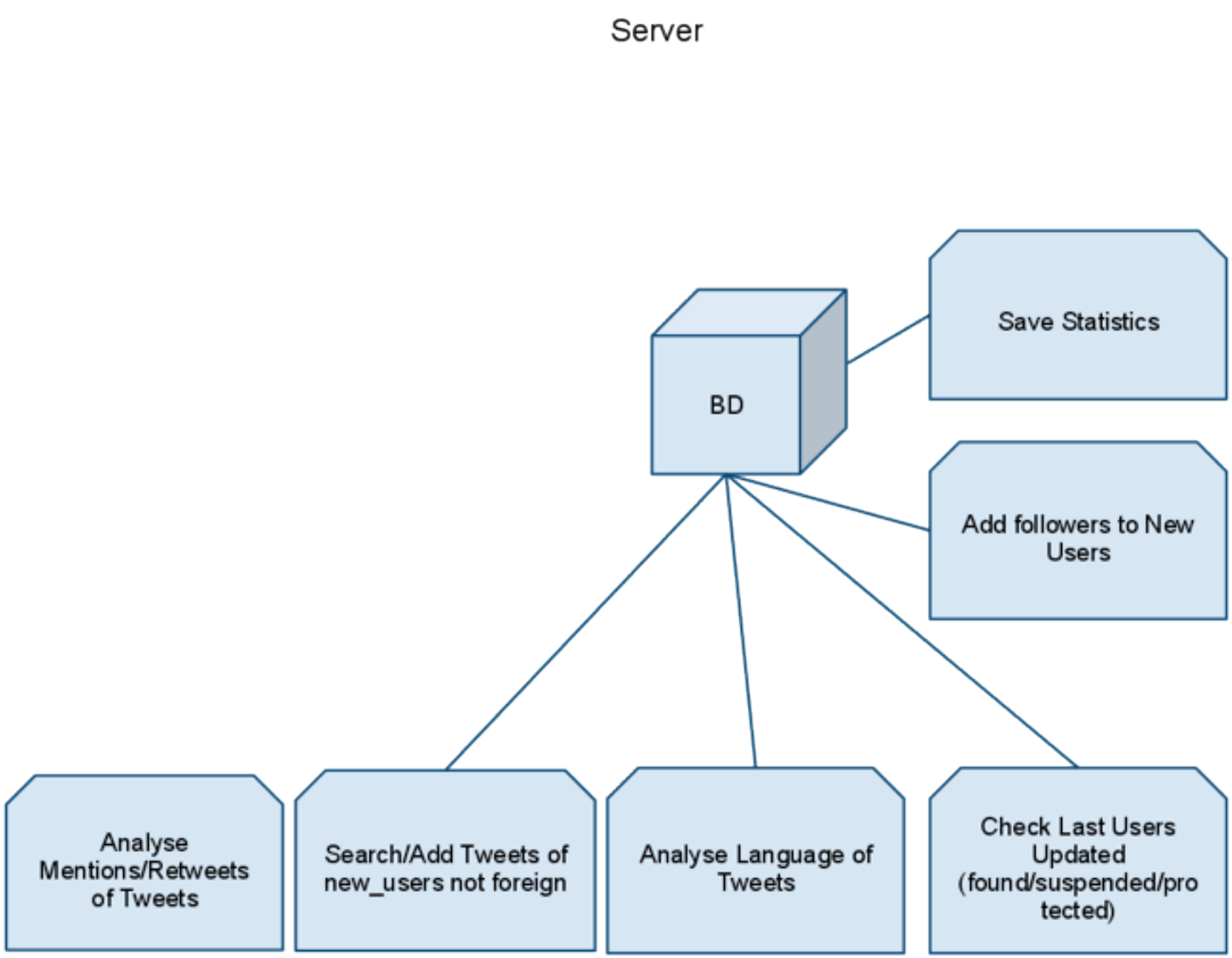
- User + Tweets Lookup
  - Gets a list of users to be looked up from Server
  - Asks Twitter API for information about users and their last tweet
  - Send JSON encoded information to Server
- Social Network Lookup
  - Gets a list of users to be looked up from Server
  - Asks Twitter API for the followers and friends of the users in the list
  - Send JSON encoded information to Server

Running in crontab

Server

- Get: gives a list of users (prioritized) to be looked up:
  - User info + Tweets
  - Social Network

- Put: parses a JSON string and inserts information in the database


Server
BD
Save Statistics
Add followers to New Users
Analyse Mentions/Retweets of Tweets
Search/Add Tweets of new_users not foreign
Analyse Language of Tweets
Check Last Users Updated (found/suspended/protected)

Server Side Processing (I)

- Starts from a set of initial Portuguese users that become the seed of the crawl
  - from TwitterPortugal
- Find new Portuguese users by analyzing Mentions and Retweets in the crawled tweets
  - tests location, timezone and name
  - uses Twitter Search API to crawl tweets to try language identification

Server Side Processing (II)

- Language Detection
  - Based on character-gram distribution
  - detects the language of the crawled tweets
  - Allow identifying Portuguese vs foreign languages.
  - For large enough samples it also allows to detect European vs Brazilian Portuguese

Server Side Processing (III)

- Checks last users updated trying to detect if the account became suspended or protected
- Gets followers of users
  - randomly inserts followers to new_users to be tested and expand user base
- Saves Crawling Statistics
  - number of rows of each table to allow error detection.

Distributing the Crawl Effort

- We have installable crawler clients (slaves)
  - Perl Modules
  - Readme files
- Can you install 2-3 clients in you Labs?
  - 24/day http access

Curating Twitter Data

Matko Bosniak

Curating Twitter Data

- Main concerns:
  - Converting crawled data to user friendly format
  - Ensuring data sanity:
    - No inconsistences, no duplicates, correct cross-refs
  - Providing chronologic navigation of data:
    - Social Network now vs 1 month ago
  - Allowing simple access to data (easy to use API)
    - Access to all fields, "Temporal" Operators, Good Performance

Curating Twitter Data: Schema

- Basic schema:
  - Users:
    - Name, Location…
    - Metadata: Is protected? Is bot? last message data….
  - Status messages:
    - User id
    - Message text, date
    - Language info
    - Is retweet / reply, from who (may require another table)

Curating Twitter Data: Schema (II)

- Pre-processed status messages:
  - Tokenized text
  - Error correction / normalization
  - Has (+/-) smileys?
  - Topic classification
- Explicit Social Network (A follows / is friend of B):
  - Edge (edge_id, A, B, type, starting data, end date)
- Implicit Social Network (A replied / retweeted B):
  - InstantEdge (edge_id, A, B, type, data)

Curating Twitter Data: Access

- Webservice API:
  - Authenticated Access to data
  - Now: a very simple API for accessing messages
- Access to dumps of the data set for offline processing

Next Steps

- Matko is just starting now (Welcome Matko!)
  - Already integrating data from Crawler
  - We will be:
    - Managing and curating the database
    - Maintaining the crawler (with Gustavo + Jorge)
    - devising and implementing algorithms for detecting influent users (per “topic”): main goal!

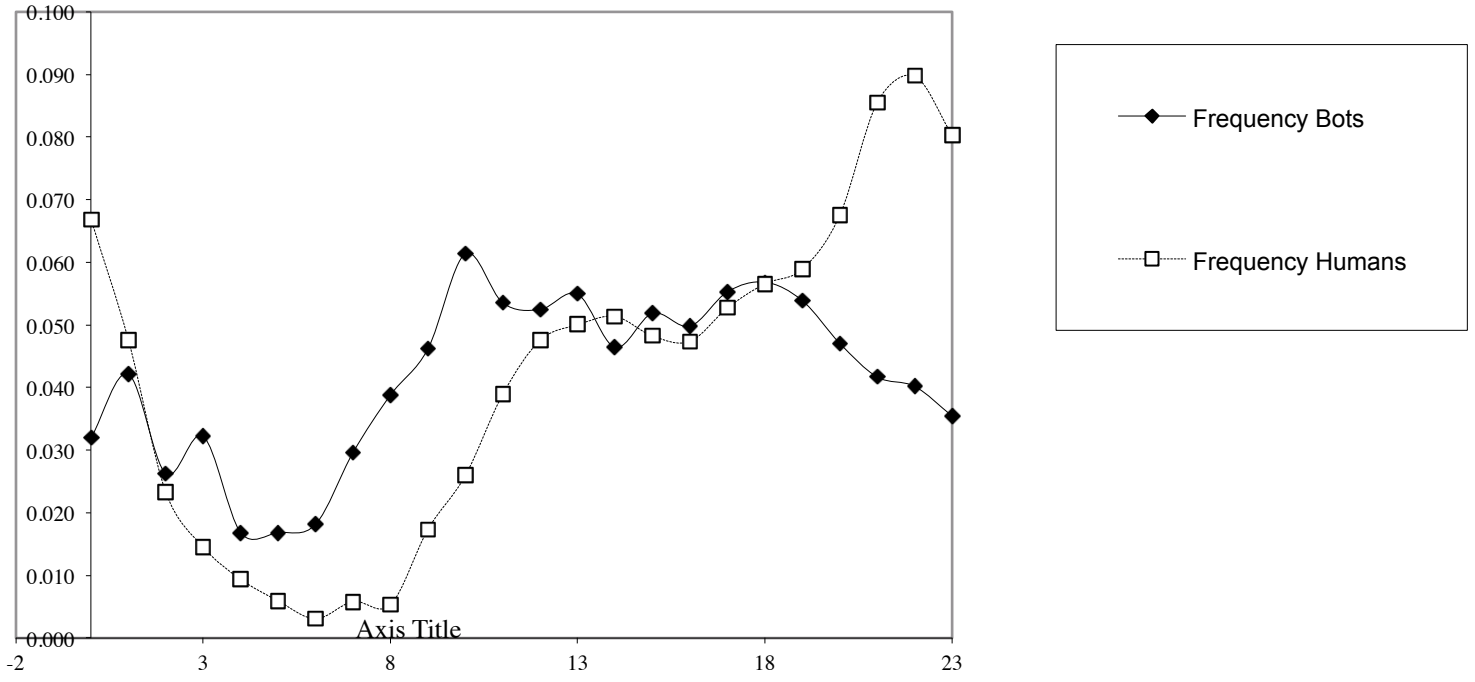
Pre-processing

Gustavo Laboreiro

“Bot” Detection

- Many Tweets are either “bots” or automatic message sending systems (“feed-like” users)
- We are training classifiers to detect these users based on:
  - Time profile of messages (periodic, nightly)
  - Size of messages (usually max size)
  - Number of links and diversity of targets
  - Syntactical features (e.g. starting label)
  - …

Bots: Illustration of behviour



Tokenization

- We already have a Tokenizer for UGC based on SVM classification
- Current developments:
  - Increase performance (it is a bit slow)
  - Use additional features
  - Have language dependent models
  - Have sub-genre dependent models (Twitter vs comments)

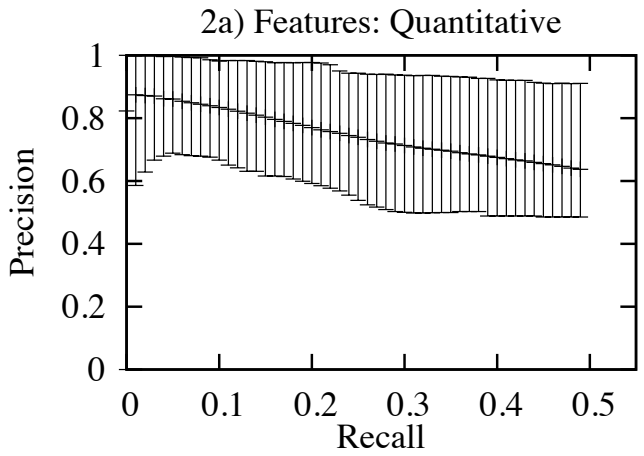
Name Normalization

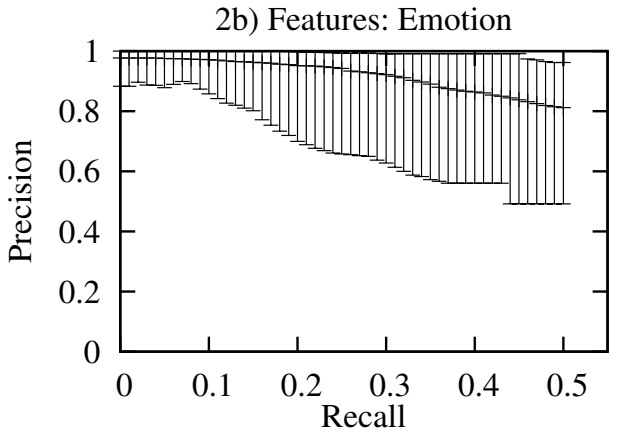
- Simple algorithms for matching partial names
  - Teixeira dos Santos – Teixeira Santos
  - In cooperation with Voxx
- Also with small variations
  - José Sócrates – Socrates
- Also trying more sofisticated matching, using context to find mapping such as:
  - Sócrates -- Pinócrates

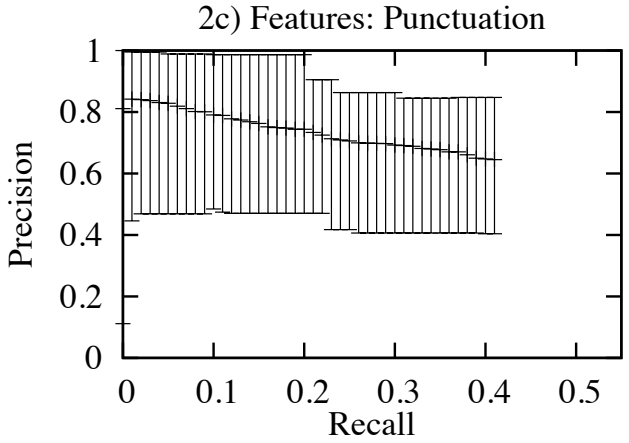
Stylistic Analysis

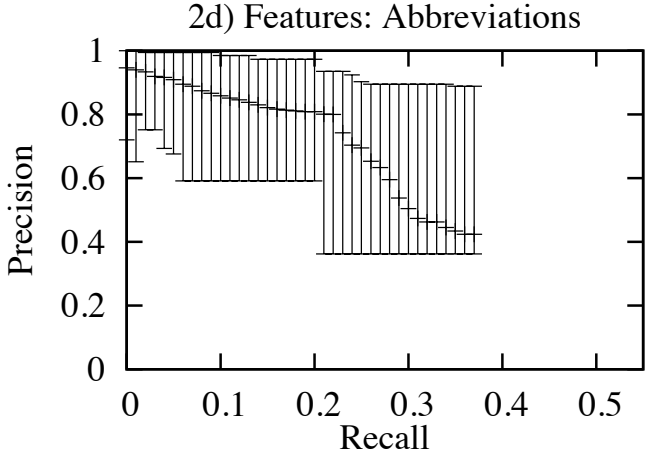
- Large set of detectors for stylistic patterns
  - Quantitative / Frequency Markers:
    - average length, number of 1-char tokens, 2-consonant tokens…
  - Marks of "Emotion":
    - Detect and classify smileys 'LOLs' and interjections
  - Punctuation:
    - "…" vs "…."; "!!!!!!"
  - Abbreviations used

Stylistic Analysis (II)









Conclusion

- We have a lot of things running
- We are mostly focusing on:
  - Collecting
  - Organizing
  - Pre-processing
- Information Extraction is left for you
  - But we can help in Robust NER